REMARKS

Claims 7, 9, 16-17 and 24 are currently amended. Claims 5, 14-15, 22-23 and 25 are canceled. Claims 1-4, 6-13, 16-21, 24 and 26-37 are pending in the instant Application.

Claim Rejection – 35 U.S.C. §102

Claims 9-13 and 15 are rejected under 35 U.S.C. 102(b) as being anticipated by U.S. Patent No. 5,193,191 ***(McKeeman)***. Applicants respectfully traverse this rejection.

Claim 9

For ease of illustration, amended claim 9 is discussed first. Claim 9 as amended recites, *inter alia*, "wherein the instructions that when executed enable the processor to initiate compiling of the file based on determining that the file was modified comprise instructions that when executed enable the processor to indicate in a work queue that the file has been modified and to initiate compiling of the file in response to detecting the indication." In the Final Office Action with respect to claim 7, the Examiner argues that ***McKeeman*** teaches this claimed feature because ***McKeeman*** discloses a "journal" for keeping track of code segments that do not need to be re-compiled. *See* Final Office Action, p.8-9 (citing ***McKeeman***, col. 11, ll. 44-61). As quoted, ***McKeeman*** teaches that code which is **not** changed/modified is indeed **not** recompiled. That is, ***McKeeman*** discloses that once a section of code is compiled, that code does not have to be compiled again if it has not since been modified. In contrast, claim 9 calls for the processor to indicate in a work queue that the file has been modified and to initiate compiling of the file in response to detecting the indication. This is clearly different from the teachings of ***McKeeman***.

Moreover, a "journal," as taught in ***McKeeman***, is **not** a work queue. The "journal" does not act as a queue for files upon which work needs to be done. Rather, the "journal" saves information indicative of past activity within the source code and "if the input has not changed, and certain other validity checks are passed, the contents of the journal will be the same as what would pass [from the compiler]." *See* ***McKeeman***, col. 11, ll. 51-61. In fact, the "journal" may save portions of code that do <u>not</u> need to be compiled. Thus, by its very operation, the "journal" is not a work queue. Further, there is no teaching in ***McKeeman*** that suggests all code that does not need to be compiled is thus marked in the "journal." That is, the "journal" is not a complete, exhaustive list of code that does not need compilation, nor does the "journal" provide a queue or list of files that do need compilation, as called for in claim 9. The "journal" does not simply rely on changes in inputs to the code, but on additional "other validity checks". Such a teaching in ***McKeeman*** shows that the described "journal" is not a <u>work queue</u>, and shows that the "journal" is not indicative of modified files needing compilation. .

The Examiner appears to be arguing that keeping track of code that does *not need compiling* is equivalent to <u>the processor indicating in a work queue that the file has been modified and to initiating compiling of the file in response to detecting the indication</u>. *See* Final Office Action, p.11. Such a position is incorrect at least because there is no disclosure in ***McKeeman*** (implicit or otherwise) to support the Examiner's assertion. In other words, in view of the teachings in ***McKeeman***, a list of code portions that do not need to be compiled is **not** a <u>work queue</u> as argued by the Examiner. Indeed, ***McKeeman*** makes no such teaching, and this is not surprising because ***McKeeman*** is not concerned with work queues.

For at least these reasons, claim 9 and its dependent claims are allowable. For similar reasons, the remaining claims are also allowable.

For ease of illustration and organization of arguments, any further remarks relevant to claims 9-13 and 15 are made in the claim 1 arguments section below.

Claim Rejection – 35 U.S.C. §103

The Examiner rejects claims 1-4, 6-8, 16-21, 23-29, 31-32 and 35-37 under 35 U.S.C. §103(a) as being unpatentable over ***McKeeman*** in view of "Upgrading Microsoft Visual Basic

6.0 to Microsoft Visual Basic .NET" ***(Robinson)***. Applicants respectfully traverse this rejection.

Claim 1

For ease of illustration, amended claim 1 is discussed first. Claim 1 calls for initiating compilation of a file in a processor-based system in advance of a request from a user to compile the file and detecting the user request to compile the file. Claim 1 also calls for indicating a status of the compilation of the file in response to detecting the user request. Initiating compilation of the file includes compiling the file in response to determining that the file has been modified, identifying the modified file in a work queue and initiating the compilation of the file based on the modified file being identified in the work queue.

The Examiner's rejection of claim 1 is improper because ***McKeeman*** and ***Robinson***, either alone or in combination as cited by the Examiner, fail to teach all of the claimed features. For example, as discussed above with respect to claim 9, claim 1 recites initiating compilation comprises identifying the modified file in a work queue and initiating the compilation of the file based on the modified file being identified in the work queue. As discussed above, ***McKeeman*** does not teach this claimed feature. Additionally, ***Robinson*** does not remedy the fundamental deficiencies of ***McKeeman***. ***Robinson*** is concerned with features of Visual Basic .NET, but ***Robinson*** is silent with respect to the claimed feature of identifying the modified file in a work queue and initiating the compilation of the file based on the modified file being identified in the work queue. Therefore the combination of ***McKeeman*** and ***Robinson*** fails to teach or suggest identifying the modified file in a work queue and initiating the compilation of the file based on the modified file being identified in the work queue, as called for by claim 1.

Further, without using improper hindsight reasoning and using the claim as a roadmap, the person of ordinary skill in the art would have no apparent reason to modify the references to arrive at the subject matter of claim 1. The Examiner has essentially provided a conclusory statement that adding the features of these references together would make for a better product; *i.e.*, the Examiner has simply stated the result of such a combination. *See, e.g.,* Final Office Action, pp. 5-6. The Examiner argues that a time saving tool would have been instantly recognized in the art. *See id.* That is, it appears the Examiner has taken the position that simply because a feature/teaching is in a reference, this is motivation enough to combine. Applicants respectfully disagree and maintain that the Examiner has conclusorily stated that such a combination would have been obvious. The Examiner has not pointed to any teachings in the cited references that would **motivate** a person of skill in the art to combine the references. In other words, the question that must be addressed includes "***why*** would a person have thought to combine the cited references based on their teachings?", and "***what*** was the need?", not simply "what benefits would result?". There must be some motivation or need as to why a combination would have been obvious at the time of the invention.

Applicants respectfully submit that the Examiner's conclusory statement is motivated by improper hindsight and is without support. Applicants respectfully request that the Examiner provide a motivation to combine/substitute that **does not** rely inherently upon the result of such a combination. In other words, a conclusory statement that that "when coupled [the cited references] would teach the claim limitations" is without proper basis and relies entirely upon the result to provide motivation. Applicants respectfully request the Examiner point to a teaching the cited art that shows **where** and **why** a person of skill in the art would have had a need to combine/substitute. In light of the fact that ***Robinson*** specifically discusses parsing and ***McKeeman*** is not concerned with background compiling, the Examiner must show some need for background compilation, not merely a result-oriented statement. Motivation to combine aside, as discussed above, even if ***McKeeman*** and ***Robinson*** were to be combined, claim 1 as a whole would be untaught and non-obvious over the references.

For at least the aforementioned reasons, claim 1 and its dependent claims are allowable. For at least similar reasons, the remaining independent claims, and their respective dependent claims are also allowable (including claim 9 and its dependent claims).

As such, Applicants respectfully request that the rejection of claims 1-4, 6-13, 15-21, 23-29 and 31-37 under 35 U.S.C. §103(a) be withdrawn.

Claim 30

Claim 30 is rejected under 35 U.S.C. 103(a) as being unpatentable over ***McKeeman*** and ***Robinson***, and further above in view of U.S. Pat. Pub. No 2005/0108682 ***(Piehler)***. Applicants respectfully traverse this rejection.

Claim 30 depends indirectly from independent claim 24. Because ***McKeeman*** and ***Robinson*** fail to disclose all of the features of claim 24 (for at least the reasons discussed earlier), these references likewise fail to teach the features of dependent claim 30. For at least this reason, claim 30 is allowable.

Claim 33

Claim 33 is rejected under 35 U.S.C. 103(a) as being unpatentable over ***McKeeman*** and ***Robinson*** and further in view of ***Callahan, II.*** Applicants respectfully traverse this rejection.

Claim 33 depends indirectly from independent claim 24. Because ***McKeeman*** and ***Robinson*** fail to disclose all of the features of claim 24 (for at least the reasons discussed earlier), these references likewise fail to teach the features of dependent claim 33. For at least this reason, claim 33 is allowable.

Arguments with respect to other dependent claims have been noted. However, in view of the aforementioned arguments, these arguments are moot and, therefore, not specifically addressed. To the extent that characterizations of the prior art references or Applicants' claimed subject matter is not specifically addressed, it is to be understood that Applicants do not acquiesce to such characterization.

Claim 34

Claim 34 is rejected under 35 U.S.C. 103(a) as being unpatentable over ***McKeeman*** and ***Robinson,*** and further in view of prior art of record U.S. Pub. 2005/0114771 ***(Piehler)***. Applicants respectfully traverse this rejection.

The Examiner's rejection of claim 34 is incorrect at least because ***McKeeman***, ***Robinson,*** and ***Piehler***, alone or in any combination, do not teach at least one of the claimed features. For example, claim 34 recites suppressing at least one of an error and warning that is detected while compiling the modified source files. In the Final Office Action, the Examiner admits that ***McKeeman*** does not teach suppressing at least one of an error and warning that is detected while compiling the modified source files, as recited in claim 34, but rather that ***McKeeman*** teaches disclosing an indication of errors. The Examiner goes on to argue that ***Piehler*** discloses suppressing errors because ***Piehler*** allegedly teaches error recovery and correction. *See* Final Office Action, p.18 (citing ***Piehler***, ¶¶[0034]-[0036]). Even assuming *arguendo* that ***Piehler*** makes such a teaching, the combination of ***Piehler*** with ***McKeeman*** is not proper. The Examiner argues that "[i]t would have been obvious to one of ordinary skill in the art at the time the invention was made to use McKeeman's error indication with Piehler's error suppression in order to make code completion more robust as suggested by Piehler (see paragraph [0035])." *See* Final Office Action, p.18. Applicants respectfully submit that the Examiner's reasoning is self-contradictory; it is unclear how it would have been obvious to use error indication with error suppression. That is, error indication and error suppression are opposite concepts. The Examiner argues that such a combination would "make code completion more robust as suggested by Piehler (see paragraph [0035])," but the alleged ***Piehler*** motivation (cited from ***Piehler***) is only in the context of error suppression, **not** error indication. In other words, ***Piehler*** does not teach or suggest that error indication provides for a more robust code completion. As such, it is clear that ***Piehler*** teaches away from ***McKeeman***. In other words, error suppression and error indication are not compatible concepts, and Applicants respectfully submit that it would not have been obvious to combine these incompatible concepts. For at least these reasons, ***Piehler*** teaches away from ***McKeeman*** and their combination is not proper.

For at least these reasons, claim 34 is allowable.

In the Advisory Action dated April 12, 2011, the Examiner indicated that the previously filed amendment would not be entered because they raise new issues and would require further reconsideration. Applicants respectfully assert that the amendments and arguments provided in the instant response overcome the Examiner's rejections, and thus, the pending claims of the present application are allowable.

Applicants respectfully assert that in light of the amendments and arguments provided throughout the prosecution of the present application, all claims of the present application are now allowable and, therefore, request that a Notice of Allowance be issued.

If for any reason the Examiner finds the application other than in condition for allowance, **the Examiner is respectfully requested to call the undersigned attorney** at the Houston, Texas telephone number (713) 934-4069 to discuss the steps necessary for placing the application in condition for allowance.

Respectfully submitted,

WILLIAMS, MORGAN & AMERSON, P.C.
CUSTOMER NO. 62293

Date: April 28, 2011

By: /Jaison C. John/
Jaison C. John, Reg. No. 50,737
10333 Richmond, Suite 1100
Houston, Texas 77042
(713) 934-4069
(713) 934-7011 (facsimile)
ATTORNEY FOR APPLICANT(S)